Architecture for an Open Source Land Administration Application

Geoff HAY and G. Brent HALL, New Zealand

Keywords: Open Source, Semantic Web, Temporal Database, Relational Data Model, Object-Oriented.

SUMMARY

This paper describes a free/libre open source (FLOSS) spatio-temporal land administration software tool for use in under-developed and developing nations. The application architecture defines an approach that addresses and improves upon a number of difficulties that affect existing models and approaches. These difficulties include securing agreement on domain semantics and concepts; defining clearly what is included in and excluded from the proposed model; discovering and obtaining consensus on model objects across many jurisdictions; and issues associated with translating the real world situation into a computational model that is both simple enough to implement, maintain and extend in the real-world situation where information technology skills may be less available, and complete enough to be useful and durable across jurisdictions. The proposed architecture is based on the notion that the formal processes associated with a particular domain contain the understanding (including the data model) for that domain. The architecture defines an 'instrument' as the temporal process (including associated data) that adds amendments (via events) to existing attribute or spatial data. The domain understanding is externalised in the form of domain ontology using Semantic Web technologies which allows for greater machine understanding of terms and types, and with the addition of workflow process definitions that use and maintain the domain ontology, allows for a less constrained and complete model of the real world to emerge. The proposed architecture allows for both cross-jurisdictional differences in semantics and for the evolution of data models over time, and does not require consensus on models or cross-jurisdictional differences to be harmonised before compilation. Importantly, the architecture enables the emergence of domain understanding and harmonisation of more specific models as the software is configured and used. The architecture steers deliberately away from the traditional state-based approach to modelling temporal processes by defining an event-based approach that is capable of storing attribute level history.

An initial prototype of this approach (Hay et al., 2008) is currently being extended with the integration of a workflow engine that enables the construction of specific software implementations and the extension of the cadastral domain model potentially into applications for other spatio-temporal domains.

Architecture for an Open Source Land Administration Application

Geoff HAY and G. Brent HALL, New Zealand

1 INTRODUCTION

The issue of land administration spans many spheres of activity and responsibility and involves to varying extents the activities of civic, legal, and professional participants as well as members of the lay public in the buying, selling and general conveyancing of land. Land ownership and/or occupancy, that has either a legal basis enshrined in civil law or an informal, customary arrangement engrained in traditional practices and use of land outside of any formal legal system, are cornerstones of modern democracies and traditional societies. In the former, the creation and transfer of titles or deeds have profound impacts on the level of economic development that stem from the property market. In this case, the relationship between land ownership and its associated rights, responsibilities and restrictions defines more than the intersection of people, land and property. It constitutes a social order that underpins the very nature of how society and economy interact around a defined and demarcated land base.

The importance of land and the legal and/or traditional rights, restrictions and responsibilities that are associated with titles or deeds to specific parcels, have lead most jurisdictions to implement some form of system to manage this information. Such systems may be traditional where occupancy and rights are 'understood' and passed down from generation to generation without any form of demonstrable, paper title or deed. Other systems may use a hard copy register or book of deeds or titles with cross-referencing to hard copy maps of surveyed parcel boundaries. In the digital era, it is more likely for countries to have adopted some form of digital database of land parcel ownership, which is managed by a computer software application. The transition of land ownership management from the traditional approach through to the use of modern computing technology is typically associated with the level of economic development of a nation. However, universal access to, and the widespread use of, computing and methods of digital data collection in the modern era has disrupted this relationship by catapulting many nations directly into the digital age. This has happened, either as a pull from outside through the activities of international aid agencies or a push from within, in many cases independent of the readiness of countries to embrace modern computing and to formalize the management of ownership of their land resource.

International development aid now often places a high priority on the land issue through investment programs that seek to achieve all or at least some of the goals of sustainability, improvement of tenure security and access to land, good governance through transparency and accountability in decision making, and improved public data access (Mitchell et al., 2008). These goals have served to encourage the adoption of projects that consider the institutional, policy and technical aspects of land administration. In the latter, information and communication technologies (ICT) are adopted by recipient states seeking to modernize their ICT infrastructure for the purposes of land administration as well as other objectives. As a result, traditional forms of land ownership record keeping have become transformed either through conversion of the old system to a direct replicate that is ICT-based or to the introduction of a completely new form of land records keeping. Either way, this process requires significant change to be absorbed, often in a short period of time, from the way that things were traditionally accomplished, to embrace new systems and methods of land administration. As a result, traditional workflows change often very rapidly and roles of staff within organizations, such as land titling offices, become transformed. In addition, since many countries are following the same general path as a result of the pushes and pulls noted above, there is systemic change involved. Although these changes are also applicable in more developed countries, the need for IT solutions that enable sharing of development effort and expertise in countries with less capacity to develop or fund their own solutions is an important aspect that should considered in any general software solution.

In recognizing the difficulties that are typically associated with embracing this degree of change, this paper describes a solution to land records management that is neither encumbered nor driven by an approach that suggests or promotes universalism. Rather, an approach that advocates the use of free/libre open source (FLOSS) software and that promotes a shared approach to development is described. This approach has facilitated the production of a land administration software tool for use in under-developed and developing nations for the purposes of achieving land records administration and management in which the twin goals of flexibility and replicability are paramount. The application architecture defines, addresses, and improves upon a number of difficulties that affect existing approaches. These difficulties include securing agreement on domain semantics and concepts; defining clearly what is included in and excluded from the proposed land records administration model; discovering and obtaining consensus on model objects across many jurisdictions; and resolving issues associated with translating real world situations into a computational model that is simple enough to implement, maintain and extend in a real-world situations where ICT skills are likely to be less well developed than in economically advanced nations. The intention is to produce an implementation that is complete enough to be easily useable, robust, and extensible from small countries with up to one hundred thousand current land records to those processing potentially significantly more land records per annum. Additional requirements for the architecture are that it fully supports cross-jurisdictional differences, such as differences in processes; data requirements; languages; and IT skills, while still maintaining the integrity of the software as an open source project.

The architecture described here complements the Land Administration Domain Model (LADM) of Lemmen and van Oosterom (2006; ISO, 2008)) in that the LADM is used as the basis for extension. The architecture externalises domain concepts (derived initially from the LADM) in the form of a domain ontology using Semantic Web technologies. The use of externally defined machine-processable semantics facilitates greater machine understanding of terms and types, i.e. knowledge about the domain that would otherwise be bound up in software code is defined externally in a way that can be processed at run-time to achieve the same purpose. This reduces coupling between domain specifics (types and terms), and software code. Moreover, cross-jurisdictional differences in semantics can be accommodated by the model, and data models can evolve over time without the need for modifications to software code. The architecture does not require consensus on models, or cross-jurisdictional differences to be harmonised before software compilation. Importantly, the architecture enables the emergence of domain understanding and harmonisation of more specific models as the software is configured and used. The architecture steers deliberately away from the traditional state-based approach to modelling temporal processes by defining an event-based approach to modelling the temporal aspects of the domain that is capable of storing attribute-level history.

The rationale for the work is briefly described in the next section followed by a brief discussion of the LADM. In Section 2 the conceptual model of the Open Source Cadastral and Registry (OSCAR) toolset is presented through the model implementation, the semantic data model and the temporal data model. The application itself is then reviewed in high-level terms first with a discussion of the platform and its Open Source components. The document repository is then presented along with the workflow model and the ontology plug-in. Map and spatial operations are then briefly discussed and a series of screenshots are presented to describe the functioning of the above components. Consideration as to how this software will be used in practice and an initial assessment of scalability and performance is provided in the final section. The paper concludes with a discussion of the main points and a note of future development plans.

1.1 Rationale

Less developed and developing nations need software and expertise to implement or upgrade their land records administration systems ((Törhönen, 2001; 2004). Currently, it is common for countries to be required to undertake a formal request for proposals and evaluation process before low interest loans are made available by international donor agencies or countries for national-level projects that are intended to support the creation of global goals such as those mentioned in the previous section. Reaching these goals typically is a long process and is often interrupted along the way by unforeseen events and shocks such as financial downturns or natural disasters. To complicate matters, solutions to information-based development objectives that are both strategic and national in focus tend to attract enterprise-level technology solutions that are dominated by the large, proprietary software industry.

Countries that have engaged the use of enterprise-level, proprietary IT solutions are typically encumbered by high initial and on-going technology and human capital costs. Furthermore, generic ‘out-of-the-box’ solutions always require significant customization to suit local needs. This process of local customization is typically very complex to manage as software has to be reshaped to meet local needs, or, as is often the case, local needs have to be modified to suit what is practically possible with proprietary software solutions. This process often results in systems that cost significant annual licensing and maintenance fees, contain more software than is required, offer little or no cooperation or sharing between different countries, and that run only through interfaces in English that not all users are fully fluent in. Also, proprietary software often requires extensions and additional modules to be developed to meet requirements, and this adds to the already high start-up costs. To remedy aspects of this dominance of and reliance on proprietary solutions that may not fit local needs, a FLOSS approach in the area of land records (parcel-level) management is an appropriate and desirable development alternative.

1.2 Land Administration Domain Model (LADM)

The LADM proposed by (van Oosterom et al., 2006; Lemmen et al., 2007) represents an attempt to define domain concepts and understandings of the relationships between people and land to enable the implementation of a general software system for land records management. The stated goals of this approach are consistent with those mentioned above, namely to avoid the high costs and issues associated with proprietary software and re-implementing similar functionality independently for each jurisdiction. The LADM seeks to avoid this and to provide an extensible basis for the development of locally relevant cadastral systems, as well as enabling sharing and communication across jurisdictions based on the understanding implied by the model.

The Model Driven Architecture (MDA) development approach where data models and software in-memory models must be explicitly defined before an application can be compiled, together with the nature of the domain where it is difficult to reach consensus on and harmonize differences in the relationship between people and land across jurisdictions, means that the definition of a viable universal model is extremely difficult. This results in a model that is restricted in scope and completeness and so may not achieve its stated goals (van der Molen et al., 2004).

Translating the complex real-world situation into a computational model that is both simple enough to implement, maintain and extend in the real-world situation where IT skills may be less available, and complete enough to be useful, easily useable, and durable across jurisdictions are competing goals that are not easily reconcilable. Completeness is also hindered by the nature of the domain where the agreed understanding of the real world domain semantics must first be secured over a wide range of jurisdictions. Additional model complexity is created when the real-world (including history) must be specified within the restrictive boundaries of the relational or object-oriented data model before further work can begin.

Regard should be given to the complexity of the software code that must be produced to access data conforming to the LADM, and the ease with which this software can be modified to suit particular or new requirements. Engineering and design of land record cadastres need to take account of trends in markets and especially servicing the development of and trading in more complex commodities (Wallace and Williamson, 2006). In this context, the LADM is a base model of the current known situation. Updating or extending it to reflect new understanding or new requirements will involve complex recoding of software and database modifications. Furthermore, extensions to the domain model are required to meet specific jurisdictional requirements. This would result in parallel development efforts (rather than shared effort) potentially reducing the effectiveness of developer communities.

Consideration should also be given to the software complexity associated with implementing a temporal data model over what is essentially a current state architecture, i.e. a relational database. Much of the implementation of temporal requirements will require complex software code and complex database queries to be developed and maintained. While a certain amount of code generation is possible using the MDA approach it is still required that the understanding of the domain is made explicit (using Unified Modeling Language (UML)) before code can be produced.

Recent research in the area of semantic markup and ontology for the cadastral domain argues for the use of Semantic Web technologies especially in the area of data integration (for both applying updates and cross-jurisdictional data sharing) and highlights the inflexibility of the relational model for both domain and temporal modeling (Spéry et al., 2001; Schuurman and Leszczynski, 2006). Process and temporal aspects of the domain are also the subject of recent research with much importance placed on the concurrent definition of both spatial data and associated process models (van der Molen, 2002; Albrecht et al., 2008).

The LADM does not address the need for processes (i.e. processes associated with administering land and the updating of data). This represents the main point of departure between the model described here and the LADM. It is argued in this paper that processes are the fundamental basis for administering the relationship between people and land. However, this does not preclude the use of a registry database based on the LADM within OSCAR, in fact it is anticipated that this would be supported within the process architecture. This aspect is discussed in Section 3.1.2.

Addressing these issues, the architecture proposed in this paper externalises the domain terms and concepts in the form of a domain ontology made up of resources that describe and link concepts and terms within the domain. The Resource Description Framework (RDF) (http://www.w3.org/RDF) is used to detail the associations and relationships between data items and their types. This, together with the use of a process based architecture, allows a highly dynamic data model and results in an extremely simple database design and development model. In Section 2, a conceptual data model for a general land registry and cadastral application is presented and discussed and the architecture of the prototype implementation using open source components is described in Section 3.

2 APPLICATION CONCEPTUAL MODEL

The central feature of the OSCAR conceptual data model (OCDM) is an 'Instrument', which is some kind of legal change associated with the status of a land parcel through a formal land registration administration and land surveying process. It is expected that, for any jurisdiction, there exists (or will exist) a 'legal instrument' that is a definition and specification for each kind of change that may be applied to a legal parcel – typically this exists in government policy documents. This definition would include such things as the required data that should be collected, checked, and stored; the business rules including error checking associated with the change; the authority and roles of the various agents that are involved in the change; and the various documents that are produced by or required by the instrument.

For the purposes of computerization, the legal instrument should be supplemented with the specification of the various data stores, and of the changes to data (including the maintenance of historical data) that are required in the application of an instrument. Typically the application of an instrument involves various actors that perform tasks specified by the instrument and importantly these tasks are ordered in time, for example, an official must give final approval for the change only after all data have been collected and submitted, and other agents have performed their roles (such as checking the accuracy of coordinates etc). The application of an instrument therefore, is essentially a temporal process that changes attribute data for an area of land. The details of the change are typically embodied in legal documents that form the record of the change.

The high-level OCDM for the OSCAR framework is shown in Figure 1. The Instrument class (representing the legal instrument including its associated processes) links Agents (such as people, organizations, groups) to Objects (a Register Object such as a parcel of land or a building) via Events which implement the temporal aspects of land administration. Amendments to spatial or thematic detail are referred to by Events which are grouped by both the Object and the Instrument.

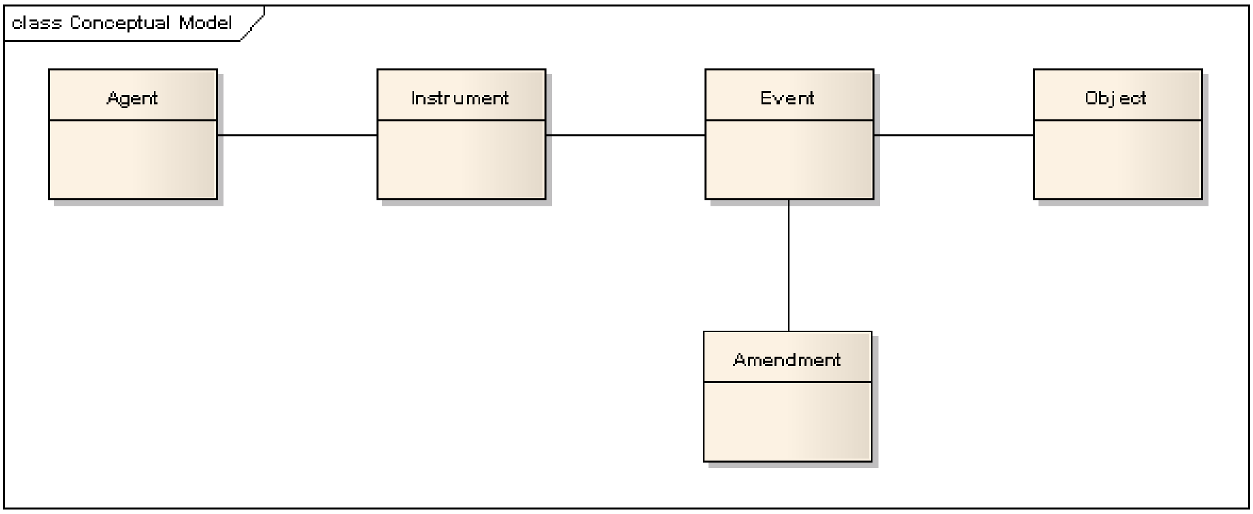

Figure 1: The OSCAR Conceptual Data Model (OCDM)

An Instrument associates ‘Agents’ such as people, banks, government agencies, conveyancers etc, as well as relevant authorities, with land and changes to the subdivision of land. An Instrument also details the rights, responsibilities and restrictions that associate land with Agents. Instruments record details about the creation, state, and change in state of the phenomenon of interest, in this case land parcels.

Classes of Instrument vary across internal and external political and national boundaries. Within a single class of Instrument (for example the issuance of a Title Document) there may be variations within national boundaries, and certainly between adjacent and disjoint countries. With all this possible variation it is unlikely that a single definition will suffice for even the simplest class of Instrument.

All Instruments used within a country therefore record and define the ontology of land administration for that country. Harmonizing the sum of instruments for every jurisdiction would reveal the complete ontology for the domain. The implementation of a single Instrument is therefore the implementation of a data model and its associated processes, and is a single aspect of the larger model. The idea of an Instrument-based data model where each instrument definition carries its own data model is made more feasible with the addition of Semantic Web technologies that allow the data models implied by instruments to be linked to a global domain ontology. Instrument-based data models also address the issues of model evolution where the data model is modified over time with new fields/relations and, cross-jurisdictional differences where the same instrument across jurisdictions may have different fields/relations.

Events associate an Instrument with the various objects that are affected in some way by that Instrument. Events detail a change to an object's state. An event also relates an Amendment (the changed aspects of the object such as its new spatial extent or changed attribute) to the object. Effectively, an object links to its current state via its events and does not explicitly store its own state. Objects (such as a parcel of land, a building area or footprint, a right of way, an easement, or some other entity) may include thematic attributes and other properties but this would be limited to those not applied by Instruments via events (and therefore are not temporal).

2.1 Model Implementation

Reflecting its inherently temporal nature, an Instrument is implemented as a workflow process that creates and modifies documents within a document repository. The documents detail the application of an Instrument including its attributes and processes. During the execution of an Instrument various event records are created that associate specific changes detailed in the Instrument, with spatial and/or attribute data. Importantly, events only provide amendments to existing data. As in other disciplines (such as accountancy), existing value data are never overwritten rather amendment records are inserted along with existing data.

The computerization of an Instrument is therefore achieved with the definition of the workflow process that manages a change to user data. Importantly, the definition of the attribute data that is to be stored, its types, allowed values, labels, integrity checks etc, may also be defined within the workflow process. Metadata such as attribute types, labels and other fields held in the workflow definition may be used in the dynamic configuration of application software user interfaces. The workflow process definition is stored as extensible markup language (XML) within the document repository. The newly changed data along with the associated log of change is stored within one or more documents that are the end result of the work process. These documents are stored as RDF/XML resources within the document repository.

Events are stored in a database table that contains minimal attribute fields that provide links (using Universal Resource Identifiers (URI)) to the Instrument type and definition, as well as the actual digital document that is the result of the application of that instrument on an object.

Events relate amendments to spatial and thematic data with objects. Spatial data records are stored in a spatial database. Since data are never overwritten, documents within the repository are never edited once completed. This means that the overhead associated with maintaining the integrity of attribute data in the face of updates (overwriting data) and deletes is not required.

2.2 Semantic Data Model

An external data model defined by RDF resources is stored within a document repository thereby decoupling domain and application semantics from the application source code. This is achieved through the use of semantic markup technologies that are currently used extensively in the digital document archive and library fields (see, for example, the Dublin Core metadata initiative – http://dublincore.org), and the emerging semantic Web (see http://www.w3.org/2001/sw/). An increasing number of countries are taking up these technologies especially in the area of government document management and archiving (e.g. Australia, New Zealand and others).

Resource Description Framework Schema (RDFS – see www.w3.org/TR/rdf-schema/) constructs are used to define inheritance and association relationships between concepts and terms in the global domain and more specific domains. Other relationship types such as equivalence are also defined. In RDF, a resource and its associations are described by triples where a resource is related to data values or other resources by named relationships. These relationships can be navigated based on the semantics associated with the relationship rather than as in the relational model’s tuple named attribute relationship, where the application relies on the explicitly predefined database data model names and types. Terms, concepts, and relationships can be added, refined, and extended over time and the application is able to handle additions and refinements to the model as required. This allows new, more specific domains to be incorporated and harmonized into the global ontology thereby supporting both cross-jurisdictional differences and evolution of models with relative ease over time.

The LADM is used as the initial set of global concepts which can be further refined and extended over time. A base set of terms and relationships are coded into the application and these include general concepts from the LADM (such as ‘legal document’) and other concepts such as, ‘document section’, and ‘document field’ which are used for display purposes.

2.3 Temporal Data Model

Time is an essential element in the administration of land ownership and its associated rights, responsibilities and restrictions. Modifications to the physical dimensions of land parcels (such as sub-division or amalgamation) and processes involving land administration (such as the issuance of titles, deeds, or plans of subdivision) are typically ordered chronologically. In the land administration domain, a central interest is in changes in the state of spatial and thematic attributes associated with land over time. Historic data associated with land including changes in the spatial extent of parcels and their ownership is of central interest among other factors in terms of being able to query the history of a land-related object and also to model and manage the processes associated with changes in the state and properties of objects temporally, spatially and thematically.

While there are many well-developed and widely used models of spatial data in use, there is no single standardized solution to modeling time within a spatial data model that captures all needs across all possible domains. Reviews of various solutions to this problem in the cadastral domain can be found (Langran, 1992; Renolen, 1997; Pelekis et al., 2004). However, database vendors implement varying generic solutions but these do not adhere to any standard and typically involve record time-stamping.

An event-based model for implementing time explicitly models events (changes in the state of the system under investigation) as first class objects, allowing an efficient and complete storage of history (Peuquet and Duan, 1995); (Claramunt and Thriault, 1995); (Chen et al., 2004); (Chen and Jiang, 2000). This kind of model not only allows queries such as “what was the state of object *x* at time *t*?” and “what changes have been made to object *x* up until times *t*,..,*t+n*?”, but also easily allows such queries as “who authorized this change?” and “what other changes are associated with this change?” An event-based model supports a wide range of possible changes to objects and with the addition of a workflow engine, event modeling can explicitly handle duration (where a change takes time to complete), lifespan (where a change has a life span after which it reverts to a previous state), or cyclical or periodic changes (where a change occurs and reoccurs in a periodic manner).

Essentially, an event is created whenever a change is made to an object (e.g. a parcel changes its ownership and/or its boundary locations and dimensions, or a deed is passed on from one owner to another) and this event links the instrument of change to the object. In the case where several objects are changed by the same instrument, an event for each object is created. The event contains links to the new state which may be the new spatial extent of a parcel or polygon record, added spatial detail to the polygon record, or additional or amended thematic attribute information added to the record. Inclusion of these needs was central to the development of the OSCAR data model.

3 APPLICATION OVERVIEW

The data model and architecture specification for the OSCAR application include support for a sharable ontology for cadastral administration, a spatio-temporal data model based on events that allow the storage and retrieval of historical data, and a plug-in architecture for highly configurable deployment and distribution that can be adapted to the specific circumstances and cadastral data management needs of any given country. This specification facilitates the development of a simple and flexible FLOSS solution that can support a widespread user and contributor base and that comprises a self-sustaining solution for land records administration.

The use of an ontology based on van Oosterom et al.’s (2006) LADM combined with semantic markup technologies allows the configuration of tailored solutions that can fit with local culture, language, and practices. Furthermore, this also enables the sharing of metadata, derivation of new local models from existing models, exchange of data in a way that preserves understanding, and search and retrieval across differing country-specific implementations. The use of these technologies means that OSCAR is in fact a platform for the development and configuration of country-specific land management systems that does not require prior knowledge (of particular country's practices or data requirements), or a complete global data model. OSCAR itself only requires prior knowledge of a small set of terms which are already defined by the LADM and transferred to the OSCAR framework.

Importantly, accommodating cross-jurisdictional differences in data requirements and semantics does not require modification to OSCAR source code. Much of the effort required to develop specific requirements is in configuring OSCAR rather than developing code. This process is essentially the translation of a legal change process into a workflow process using standard workflow modeling practices.

3.1 OSCAR Platform Open Source Components

The proposed platform is based on the Open Source Eclipse Rich Client Platform (RCP) in which users contribute to the functionality via plug-ins (small software modules). The RCP supports dynamic loading of plug-ins and therefore allows various configurations of plug-ins to be deployed supporting the production and deployment of sets of independent targeted tools from the same codebase. The notion of OSCAR configurable tools developed as plug-ins is derived from this to comprise a set of functions that cover various usage scenarios and that are configurable for and by various end-users. Hence, the OSCAR platform defines a set of extensible basic tools (detailed below) that are available via published extension points.

If required, users may choose to contribute to the software project in various ways, such as by extending a tool or tools, by creating a new plug-in that extends a default tool via extension points (and which might publish new extension points), or by submitting new classes and functions for inclusion in the OSCAR code base. The RCP allows the deployment of a fully capable development environment and OSCAR software development kit (i.e. a 'ready to run' OSCAR SDK system development environment) for users to develop their own solutions and plug-ins. The Java programming language is used and this offers a number of advantages including platform independence (write once, run anywhere); FLOSS licensing; internationalization support using externally defined language files; and large code libraries including spatial and imaging (and as used by the popular Open Source spatial uDig and GeoTools applications).

It is expected that most of the development work performed by specific domain software developers will be focused on translating actual work practices and procedures into workflow definitions and the production of code specifically relating to automation of tasks within those workflows. The translation of the actual situation into a definition is done with reference to an ontology allowing the situation semantics to be captured within the definition.

3.1.1 Document Repository

The OSCAR document repository stores documents as RDF-enhanced XML documents in a similar manner to digital libraries. Documents are RDF resources and are stored separately from spatial data and are linked to spatial data via Event tables in the spatial database (via URI). Data fields (e.g. the name of an owner, an appellation, an image, etc.) are stored in document fields which are also RDF resources (and hence reference both the global and specific understanding). Storing user data in this way provides a number of distinct advantages over traditional database storage (i.e. fields in tables) including:

- No prior knowledge is required by the software of a document, its type, its fields (and their types), the language used, or its associated constraints. This means that a database data model for documents that defines these aspects is not required and this reduces complexity in both the database and the software that deals with documents.
- Users can define any number of Instrument types and associated documents with any combination of fields and types without the need to define separate storage for these different types. Since the documents including its fields and types that represent the result of the application of an Instrument, and in fact all user-specific thematic data are RDF resources, they are able to be handled in the same way by the software. Further, users are able to version documents and fields (i.e. modify the definition of an Instrument), and this does not affect previously defined document instances or require any modifications to document storage.
- Since documents are immutable (i.e. once they are created and approved they do not change), the field values in a document are therefore implicitly time-stamped (database time) attributes by default. Document hardcopies may be produced easily and in any format required by processing the RDF/XML, using existing technology to produce Web pages, Adobe PDF files, or images. Note that updates to attribute values are added as amendments which is effectively attribute time stamping.

The document repository and its associated software user interfaces for viewing resources within the repository are combined in an open source plug-in component included with OSCAR.

3.1.2 Workflow

The JBoss Java Business Process Management (jBPM) workflow plug-in includes a process designer that allows implementation of work processes associated with Instruments. The designer allows the definition of a temporal process that involves various agents and tasks (i.e. a workflow). Tasks may be associated with agents (i.e. authorized users or groups of users) or automated with external code modules. Much of the software development work that would be required by users (such as for implementing a country-specific rule) is in the development of these external modules. The workflow installation includes a workflow server that executes processes as required, and a management interface for managing process execution.

The workflow server and designer are also used internally by OSCAR to orchestrate the development and configuration of country-specific land administration systems. For example, OSCAR defines a workflow process entitled ‘Instrument Design Process’ that orchestrates the definition of an instrument. This process is used to design country-specific Instruments and automates the construction of RDF mark-up and derivation from the global ontology, the creation of specific ontology, and the linkage between them. As a user defines the fields and attributes of an Instrument he/she is using existing metadata (from the global ontology), creating and adding to the specific local ontology, while at the same time contributing to the global ontology (adding terms from the specific local ontology), and hence providing a mapping between the global and a specific ontology. This is all automated within the predefined OSCAR Instrument Design Process.

The workflow plug-in also provides a number of other benefits ‘out of the box'. For example:

- A default Web application generator which can produce and deploy a Web application for each process if required.
- Process definitions are stored as XML or any other standard exchange format (e.g. page description language (PDL), business process execution language (BPEL)).
- A secure Web style process management interface.
- It can be configured to use legacy user organization databases to provide for authentication and secure access, as well as selection of task recipients based on authentication.

OSCAR extends the workflow plug-in by adding user-interface components that integrate (via an ontology server) the semantic markup into the workflow definition. This enables the understanding of the specific semantics to be derived from the more conceptual global ontology, and stored and accessed along with the user data. Via the ontology plug-in, the workflow extension enables the definition of new or more specific terms and concepts thereby enabling the emergence of a more complete global domain model. This emergence of the domain understanding has been called the ‘missing link’ (Albrecht et al., 2008) due to the fact that defining domain ontology suffers from the same kinds of difficulties as defining a domain data model (noted above). The notion that domain ontology can be emergent and dynamic solves many of these problems in that the need to be explicit and complete is not so vital.

The use of a more traditional database model (perhaps an extended version of the LADM) and implementation is quite possible within the OSCAR framework. A jurisdiction may choose to forgo the semantic data repository (and its associated benefits) and instead provide database access and user interface classes as a plug-ins to OSCAR or as classes integrated into a workflow definition. We hope to examine this possible development option at a later date.

3.1.3 Ontology

The ontology plug-in provides a resolution service for use by the workflow and document management plug-ins to access domain references. The ontology plug-in is the repository for shared understanding for both the OSCAR components (document repository and workflow) and the domain data. The ontology management interface allows the navigation and management of ontology data and references. The ontology plug-in also defines an ontology used internally by OSCAR which is derived from the domain ontology and is based on a small subset of the LADM. This very small ontology (less than 15 concepts) is used internally by OSCAR to help orchestrate functions and mappings between the spatial database, the document repository, and the workflow. Eventually this ontology, the global domain ontology, and all country-specific ontologies could be navigable and accessible via a Web service. The ontology plug-in uses the document repository and the same RDF resource structure for storage, i.e. concepts and terms within the ontology are also resources and hence are treated the same way as documents by the software. Note that workflow definitions are also resources within OSCAR and are also stored within the document repository.

3.1.4 Map/spatial operations

Mapping, spatial operations, and spatial data storage within OSCAR are handled by third party Open Source geospatial plug-ins including uDig, GeoTools, and PostGIS, and are used by the OSCAR prototype. Others can be added and even those already included may be replaced, depending on user preference and suitability of a plug-in to a location's needs and resources. An additional extended suite of spatial operations tools (Axios) is under investigation for inclusion within OSCAR and it is expected that these will complete the set of required spatial operations tools for the software.

4 SCREENSHOTS

Screenshots of the prototype OSCAR software reveal its basic functions. OSCAR is composed of a number of components most of which are visible in the following screenshots. As noted above, these include uDIG geospatial software; jBPM workflow software; and the document repository. The PostGIS database layer that interfaces with PostgreSQL is visible in the form of a database viewer.

Figure 2 shows the basic OSCAR interface with cadastral parcels displayed in the centre map panel (the uDig software being used as a plug-in for this purpose). The database view tab on the left shows parcel information including a parcel's documented history that is made up of events and documents. Navigation of these documents and event links is essentially navigation of the history of the parcel and can be achieved by clicking on an event to show the state of the current parcel at the time of the selected event.

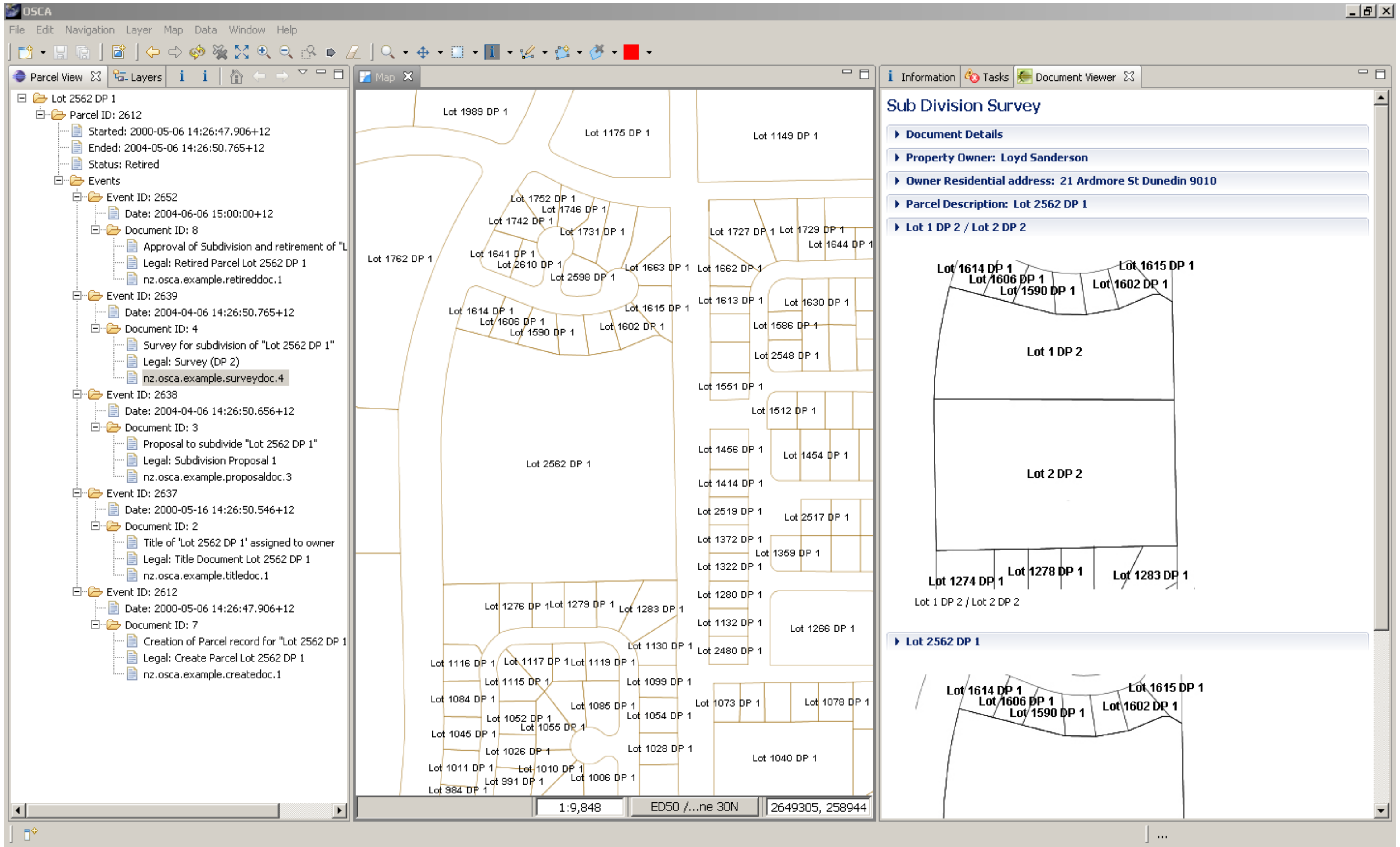

Figure 2 – Basic OSCAR User Interface

Clicking on a document link in the left panel shows that document in the far right panel of the interface. This panel is a viewer that is capable of viewing any RDF resource (a document is a resource). Within the document there are a number of links under each document field. These are the mark-up entries for that field. Each field may have any number of mark-up entries. Each entry is an RDF triple that is made up of the subject, predicate, and object parts. Each part may be a literal value or a reference to another resource which may be a concept. The entries (triples) are defined at the time the associated workflow process is created, and value data typically set as part of the execution of that process. These entries may be navigated in a similar manner to webpage links. Many different kinds of links are possible including: actual value data, data types, labels, references to images, references to other documents, references to database fields (these will show the data in the left panel), references to Web pages, references to ontology concept descriptions, execution of external software, and links to menus and task/query execution.

Figure 2 shows the history of Lot 2562 DP 1 in the left hand panel. This large lot in the centre of the map in the middle frame is subdivided into two new parcels. The original parcel is retired as part of the subdivision process. The top-most (most recent) event in the left panel is the event that retired the parcel through its subdivision and this is detailed in the associated documents. The previous event (just below the retirement event) shows the event associated with the sub-division approval and its associated document. The document which shows details of the survey is visible in the right side panel.

Figure 3 shows the retirement of lot 2562 DP 1 in the right hand panel. The retirement document details this process and refers to the creation documents for the two newly created parcels. The links inside the document can be navigated, as can links embedded in any document. Figure 4 shows the result of navigating the links in the previous document, the parcel view tab in the left panel changes to show one of the newly created parcels and shows the history of the parcel. Note that this document also has links referring to the previous parent parcel. Figure 5 shows in the document viewer the result of navigating the previous link and displays the ontology concept description in Chinese for the URI `cn.ontology.land.administration.parcel` which can be seen to be 'derived from `org.osca.ontology.concept.parcel.`

This document was created as part of a different workflow process and therefore has a different data model from the previous documents. The parcel would be part of a different jurisdiction but can exist within the same database as the other parcels shown. When the ontology description for a parcel field is selected it refers to a Chinese ontology concept that is related (perhaps by equivalence) to another domain ontology.

Figure 6 shows the workflow process designer, which is an OSCAR plug-in from jBPM that is included with OSCAR to create workflows and manage tasks. Each task node may be associated with selected users or automated with code. Tasks may involve collecting data, making decisions, or creating records.

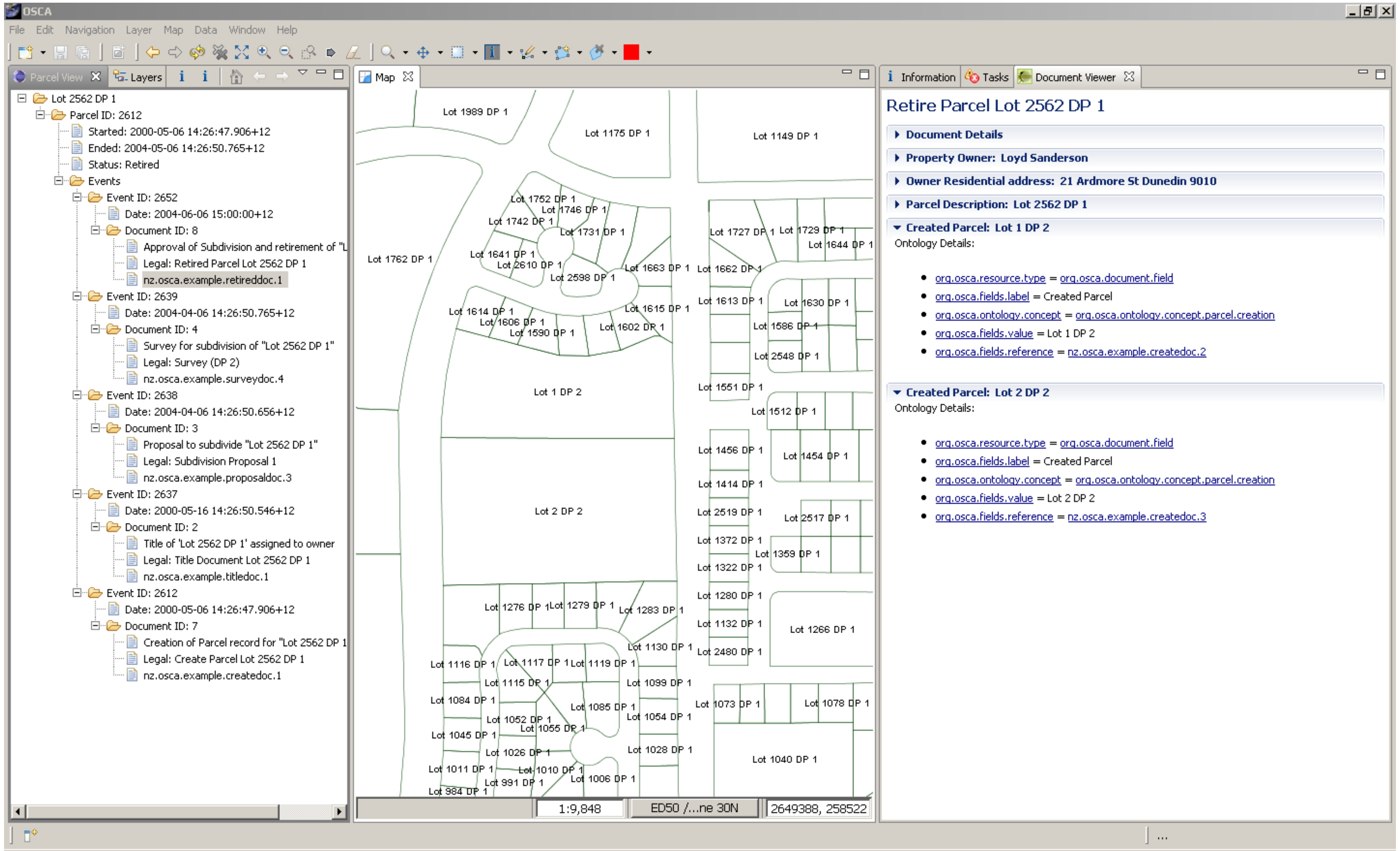

Figure 3 - Parcel subdivision and retirement

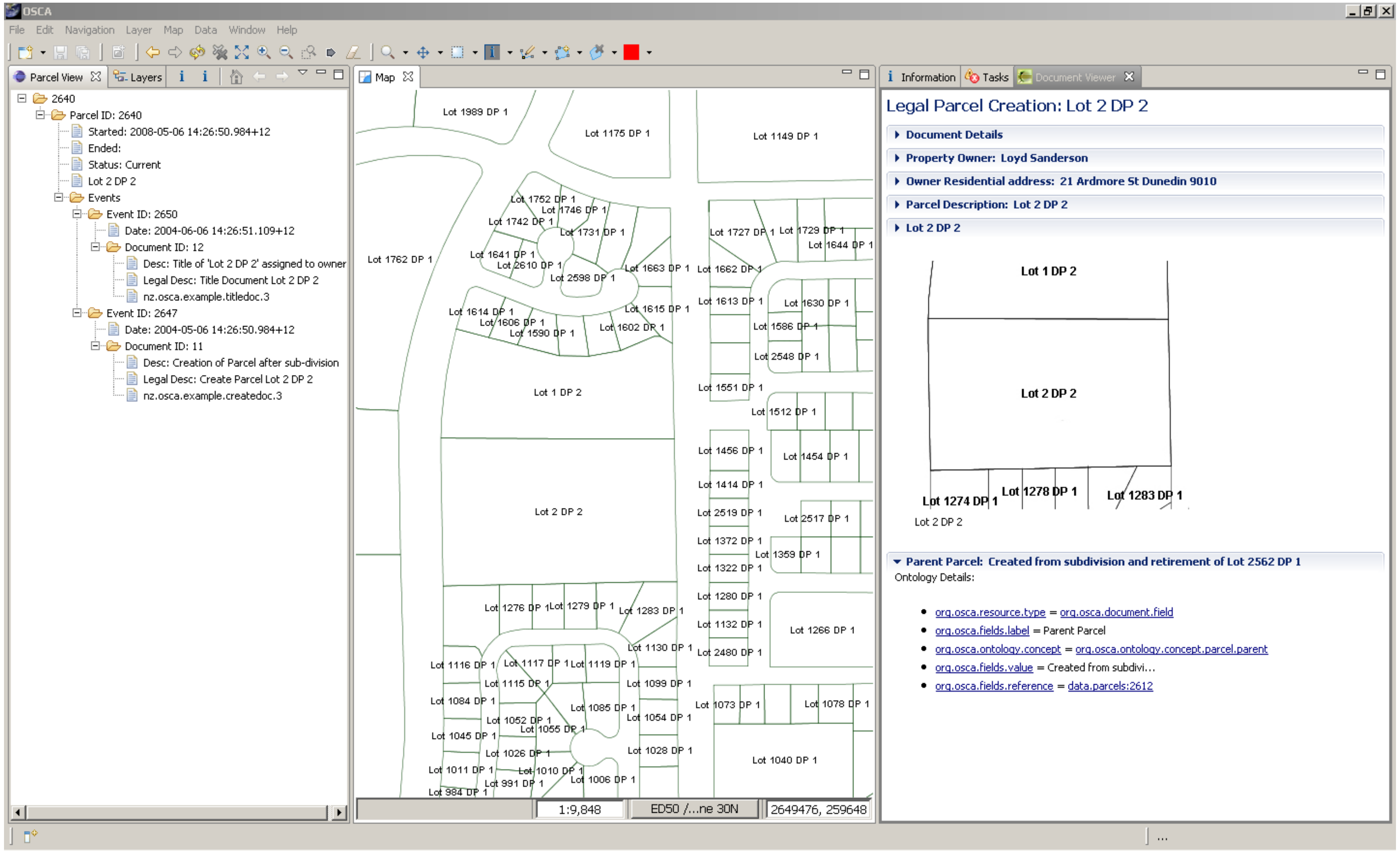

Figure 4 – Newly created child parcels

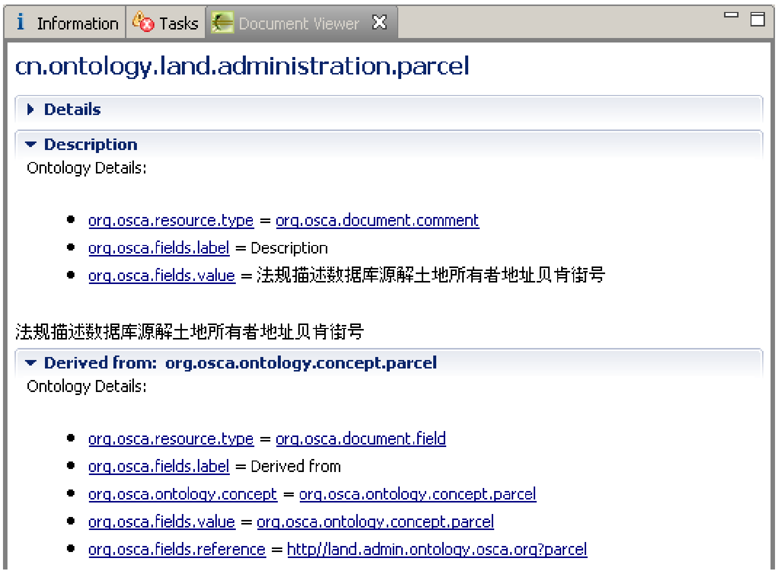

Figure 5 – Ontology markup detail

5 PRACTICE

The OSCAR framework is designed specifically to fit the needs of developing nations. Foremost, as a FLOSS project that supports multiple development and user communities, jurisdictions may choose to modify and configure the software and its plug-ins as they see fit relative to their own needs. A number of free, well understood, stable, and mature open source plug-ins have been included within OSCAR (e.g. UDig, PostGIS, PostgreSQL and jBPM. These plug-ins have large, well established and active development communities of their own and so it is expected that this will provide stability and durability to the OSCAR software and support to OSCAR users and developers.

The process-based architecture means that for most purposes, jurisdictions will be mostly concerned with the configuration of workflow processes (using the graphical workflow designer plug-in and extensions) to suit their particular needs. Typically, jurisdictions need only develop Java code snippets to implement such things as business rules and automation of legacy code (such as access to legacy databases), for which the user community could provide support and code sharing. Most of the effort in deploying and modifying the software for use in a specific setting is therefore in the translation of existing instruments (perhaps from government policy documents) into computational form using the workflow designer's graphical interface. It is not required that jurisdictions completely automate an instrument, in fact, jurisdictions may initially choose to use basic definitions that only support secure data entry, so supporting their current practices rather than replacing them. More completely automated instruments may be defined as more sophistication is required.

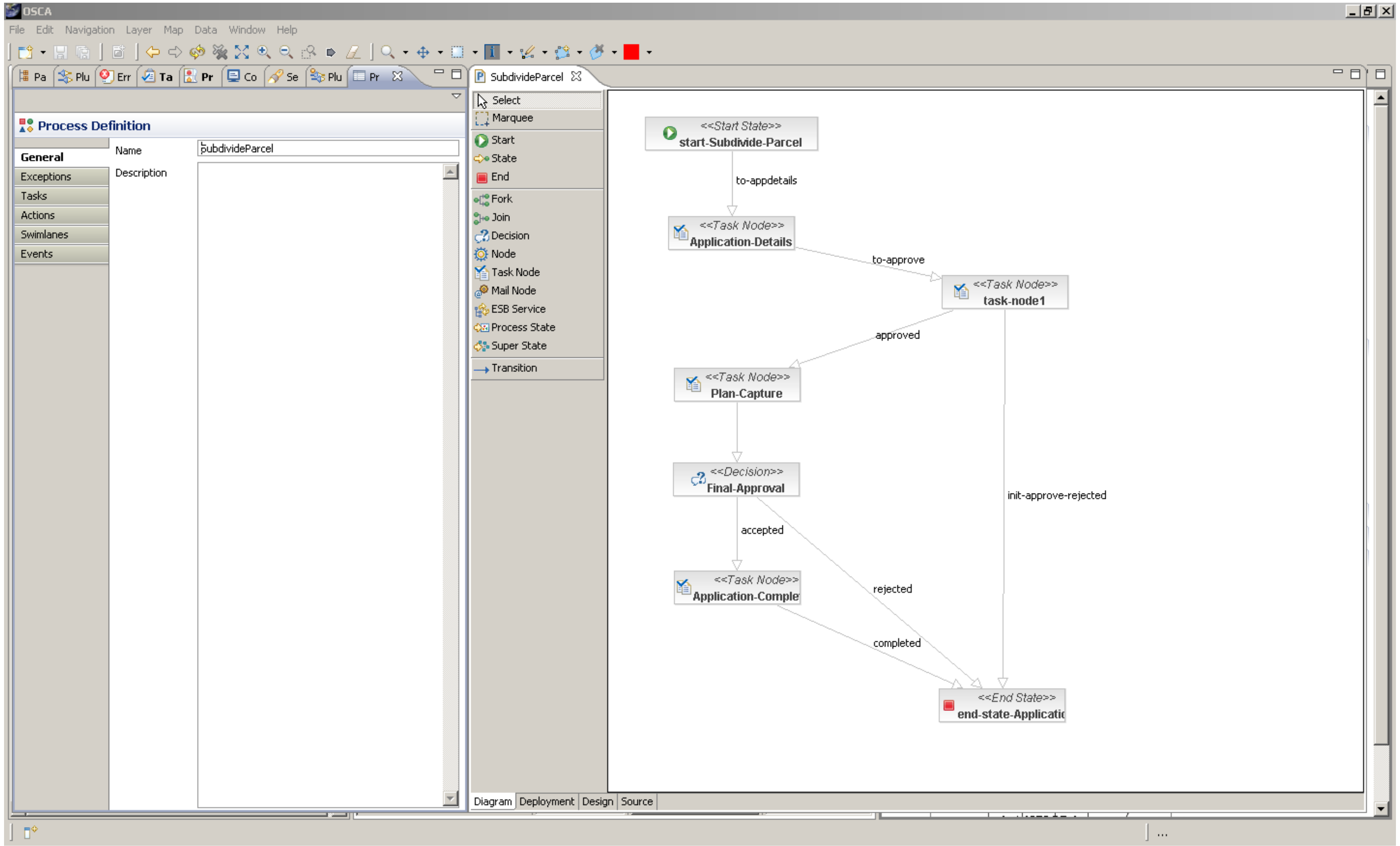

Figure 6 – Workflow process designer plug-in

It is expected that in full beta release OSCAR will contain a set of base workflow definitions that implement generic processes (e.g. data upload from a legacy database, parcel creation, subdivision, retirement, data archiving, document production, querying, indexing etc) which may be included as sub-workflows within new more specific workflows, or directly modified to suit particular needs.

5.1 PERFORMANCE AND SCALABILITY

Although OSCAR is still in the early stages of development with much work to be done especially in the integration of the workflow designer and the RDF repository, it is possible to make some broad speculations regarding performance and scalability. It is expected that more objective benchmarking and testing will be possible in the near future with work currently in progress to upload the Western Samoa cadastre into OSCAR.

Since the semantic technologies employed in OSCAR were originally developed for searching and indexing data that exist on the World Wide Web, it is expected that these technologies are highly scalable. In terms of storage, it is expected that the extra mark-up stored along with value data, will comprise a significant proportion of the total storage space but this is yet to be measured with real data. It should be noted that the extra data associated with the mark-up, is in fact also important value data that add significant durability and data integrity to the cadastre itself.

In terms of performance, the underlying platform (Eclipse RCP) and additional plug-ins (Postgre/PostGIS, jBPM etc) are optimized for performance while still allowing a high degree of flexibility. It is expected that indexing and searching operations over the document repository will take more time than similar operations over more traditional data stores. However, it should be noted that, especially for large indexes (i.e. those that index the entire cadastre for a particular point in time), once created they never need updating since data are never modified. This applies to large indexes such as an index that might be used to search the entire cadastre for particular historical values (e.g. the contact details of each parcel owner at a particular point in the past).

6 DISCUSSION

The data model and software architecture described in this paper is conceptually and architecturally more flexible in its design than the than more traditional database and client application architecture that has characterized land records management systems in the past. The architecture offers promise for a general solution to records management in developing nations that promotes sharing of development effort and domain knowledge and fully supporting differences in jurisdictional requirements, practices and customs. It does not require significant technical expertise to deploy and configure the software. It specifies a prototype application developed using mature FLOSS components in support of a data model which supports storage of historical data. The model and operational prototype are both shown to be suitable for further implementation to support the ultimate objective of a freely distributed cadastral management software package that satisfies the objectives of a land records management approach stated in Section 1.

The OSCAR framework comprises a highly configurable tool. In terms of software code, it requires very little prior knowledge (low coupling) of specific data and process requirements (i.e. a fully specified data model and user interface code that refers to specific database attributes) other than that detailed in the conceptual model (OCDM). This is achieved through the use of semantic markup technologies that are currently used extensively in the digital document archive and library fields and the emerging semantic Web.

OSCAR is designed based on the proposition that Instruments (processes and associated data), whether formal or informal, are the central feature of the link between people and land. This translates into a relatively easy deployment path for jurisdictions because much of the deployment effort is associated with translation and configuration of specific processes rather than developing software code for user interfaces that access databases (as in more traditional systems).

In summary, OSCAR which utilizes a number of standard well-understood technologies and components in ways that are innovative and highly configurable to suit different circumstances and needs. These innovations provide a number of highly desirable features in the OSCAR framework including:

- A simple database model that is easily understood, does not require modification, and only requires typical maintenance. OSCAR stores thematic data within a separate document repository, thus the required database is lightweight and only used for linking spatial, temporal, and thematic data.
- Temporal data are maintained both at the transaction level (via events) and at the attribute level (via amendments), providing a historic record for all thematic data.
- OSCAR is a data independent platform. As noted earlier, it does not require prior knowledge of user data and is therefore highly extensible to meet varying data and process requirements while at the same time still leveraging the benefits of FLOSS approaches (code sharing, development cooperation, etc.).
- Domain terms, concepts, and understandings are shared and unified in a way that enables integration and search of data across boundaries regardless of attribute names, labels, or language. Data and data labels are not required to be in any particular language.
- A new Instrument, document or attribute definition does not require any modification of the software or database, and this includes modifications to existing definitions (such as the addition of a new attribute).
- The domain ontology is emergent and dynamic. As users implement specific instruments, they are reusing and contributing to the shared global domain. As noted above, the starting point for this global domain is the LADM.

Finally, the architecture for 'Instrument-Based' data models and applications (such as OSCAR) is applicable over a broad range of application domains. Although this paper has focused on the land records management and administration, or cadastral, domain, it is equally applicable to any process oriented activity that spans both space and time. Although the OSCAR model and application as described above is developed in prototype form, the test of its suitability to real world circumstances is yet to be fully tested. Work is underway to apply the model to the case of the digital cadastral database in Western Samoa and other test cases are actively being sought by the authors. As the project continues to evolve the robustness and scalability of the application will be tested using standard performance benchmarking approaches. It is expected that the software will be made freely available in beta form for evaluation in late 2009.

ACKNOWLEDGEMENTS

We would like to thank Mike Leahy, Dr. David Goodwin, Don McKinnon and Gertude Pieper for input to this work and to thank Dr. Mika Törhönen of the Land Tenure Group of the Food and Agricultural Organization (FAO) of the United Nations for his support. The FAO provided financial support for the project.

REFERENCES

Albrecht, J., Derman, B., Laxmi, R., 2008. Geo-ontology Tools: The Missing Link. Transactions in GIS 12, 409-424.

Chen, J., Jiang, J., 2000. An Event-Based Approach to Spatio-Temporal Data Modeling in Land Subdivision Systems GeoInformatica 4.

Chen, J., Jiang, J., Yeh, G.-o., 2004. Designing a GIS-based CSCW system for development control with an event-driven approach. Photogrammetric engineering and remote sensing 70.

Claramunt, C., Thriault, M., 1995. Managing Time in GIS: An Event-Oriented Approach, Proceedings of the International Workshop on Temporal Databases: Recent Advances in Temporal Databases, Springer-Verlag.

Langran, G., 1992. Time in Geographic Information Systems. Taylor & Francis, London.

Lemmen, C., Augustinus, C., van Oosterom, P., van der Molen, P., 2007. The Social Tenure Domain Model - Design of a First Draft Model, Proceedings: FIG Working Week, Hong Kong.

Pelekis, N., Theodoulidis, B., Kopanakis, I., Theodoridis, Y., 2004. Literature review of spatio-temporal database models. The Knowledge Engineering Review 19, 235-274.

Peuquet, D.J., Duan, N., 1995. An event-based spatiotemporal data model (ESTDM) for temporal analysis of geographical data. International Journal of Geographical Information Science 9, 7 - 24.

Renolen, A., 1997. Temporal Maps and Temporal Geographical Information Systems (Review of Research). Dept. of Surveying and Mapping (IKO) The Norwegian Institute of Technology.

Schuurman, N., Leszczynski, A., 2006. Ontology-Based Metadata. Transactions in GIS 10, 709-726.

Spéry, L., Claramunt, C., Thérèse, L., 2001. A Spatio-Temporal Model for the Manipulation of Linege Metadata. GeoInformatica 5, 51-70.

Törhönen, M.-P., 2001. Developing land administration in Cambodia. Computers, Environment and Urban Systems 25, 407-428.

Törhönen, M.-P., 2004. Sustainable land tenure and land registration in developing countries, including a historical comparison with an industrialised country. Computers, Environment and Urban Systems 28, 545-586.

van der Molen, P., 2002. The dynamic aspect of land administration: an often-forgotten component in system design. Computers, Environment and Urban Systems 26, 361–381.

van der Molen, P., van Oosterom, P., Lemmen, C., 2004. Remarks and Observations related to the further development of the Core Cadastral Domain Model, Joint 'FIG Commision 7' and 'COST G9' Workshop on Standardization of the Cadastral Domain, Bamberg, Germany.

van Oosterom, P., Lemmen, C., Ingvarsson, T., van der Molen, P., Ploeger, H., Quak, W., Stoter, J., Zevenbergen, J., 2006. The core cadastral domain model. Computers, Environment and Urban Systems 30, 627-660.

Wallace, J., Williamson, I., 2006. Developing cadastres to service complex property markets. Computers, Environment and Urban Systems 30, 614–626.

BIOGRAPHICAL NOTES

Geoff Hay

Geoff graduated from the University of Otago with a first class honors degree in software engineering. He has researched in the areas of Semantic Web, agent-based computing, workflow and process modeling, database design and software metrics. He has been a GIS researcher for a number of years since and has published papers in geospatial aspects of population health. Geoff has recently started a PhD in the area of cadastral information systems with the School of Surveying at the University of Otago, New Zealand.

Dr. Brent Hall

Brent Hall completed his PhD at McMaster University, Canada in 1980. He has held academic appointments in Canada at the University of Guelph, Wilfrid Laurier University, and the University of Waterloo. He has also worked at the Universities of Auckland and Otago in New Zealand. He is currently Dean of the School of Surveying at the latter. In addition Dr. Hall has held the Belle van Zuylen Chair at the University of Utrecht in the Netherlands, been a CIES Fellow in Peru, and an Erskine Fellow at the University of Canterbury in New Zealand. His area of interest is geographic information systems, in particular developing spatial decision support software in developing countries. His current research is on the use of Open Source geospatial tools to facilitate multi-participant inputs to spatial decision problems.

Dr. Hall has co-authored one textbook, edited another, and written numerous book chapters and over sixty papers in peer reviewed journals. He is recipient of the Horwood Critique Prize from the Urban and Regional Information Systems Association, of a University-wide award in teaching excellence at the University of Waterloo, and of a national award for excellence in teaching Geography from the Canadian Association of Geographers.

CONTACTS

Mr Geoff Hay
GIS Researcher/PhD Student
School of Surveying
University of Otago
P.O. Box 56
Dunedin
NEW ZEALAND
Email geoff.hay@otago.ac.nz

Dr. G. Brent Hall
Professor and Dean
School of Surveying
University of Otago
P.O. Box 56
Dunedin
NEW ZEALAND
Email brent.hall@otago.ac.nz